Abstract: We propose a model for differential-delay compensation and two approaches for provisioning survivable service on Data-over-SONET/SDH networks with virtual concatenation. Our approaches optimize the resource subject to the constraint of differential-delay compensation.

© 2006 Optical Society of America

OCIS codes: (060.4250) Networks; (060.4510) Optical Communications

1. Introduction

Virtual concatenation (VCAT) and link capacity adjustment scheme (LCAS) technologies will enable increased management and transport capabilities for SONET/SDH enabled data services. Utilizing encapsulated generic framing procedure (GFP) and VCAT, nodes will support new service such as Ethernet, Fiber Channel, FICON, and ESCON on WDM mesh or ring networks [1]. The key to the deployment of this new technology is that it will only affect nodes at the add and drop points and will not affect the intermediary SONET/SDH links. These emerging next-generation SONET/SDH techniques, which are supported by advanced framer/mapper devices, deliver many benefits and efficiency to optical networks by providing multi-service provisioning platform (MSPP) across legacy backbone infrastructure [1-3]. However, the independently routed VCAT group members cause differential delay at the destination node. This differential delay, which is limited by the end node compensation capability, network OAM, or end application QoS, can impact service if not accounted for in the routing.

2. Differential Delay in Data-Over-SONET/SDH Networks

Since the VCAT group can be routed independently over different paths, the frame can arrive at different time at the receiving end. Due to this asynchronous nature, the receive equipment needs enough memory to buffer the frame locally until all the frames that belong to the same group arrive so as to reassemble the payload to its original form. We propose a simple but practical differential delay model when provisioning traffic over multiple paths. Our model is simple enough for real-time implementation. Generally, both diverse and non-diverse routed VCAT group (VCG) members can cause differential delay (DD) since even if the VCG are transmitted along same route and via same fiber, it may happen that the signals are transmitted on different wavelengths. In the non-diverse routed case, the DD is composed of deterministic and non-deterministic DD. The deterministic DD is constant for the duration of a connection and it’s composed of optical delay spread and equipment delay. In Ethernet-over-SONET/SDH service, MAC switching functions integrated into the overall point-to-point service lead to emission and queuing delay [4]. In summary, although there are many factors, the DD is primarily attributed to propagation delay. Without loss of generality, we model the DD as propagation delay and network equipment delay. The propagation delay = distance * (refraction index) / (speed of light). For example, across the continent from east to west, the delay is 5000km * 1.46/(2.998E5 km/s) = 24.34 ms. In industry approximation, the propagation delay is 5 µs/km. Network equipment delay includes ADM and Time Slot Interchange (TSI) for cross-connect. In generic criteria, based on [5], the maximum latency is 25 µs.

So, given a path P, we can model the packet delay $D_p$ as the following:

$$D_p = \sum_{(i,j)\in P} d_{i,j} \cdot r_1 + \sum_{v\in P} v_j \cdot r_2$$  \hspace{1cm} (1)

where \(r_1 = 5 \text{ µs/km}, \ r_2 = 25 \text{ µs}.\ d_{i,j} \) is the length (km) of link \((i,j)\). \(v_j = 1 \) if node \(v_j \in P\), \(v_j = 0\), if \(v_j \notin P\).

Differential delay (DD) between path $P_1$ and $P_2$ is:

$$DD_{(P_1,P_2)} = |D(P_1) - D(P_2)| = \left( \sum_{(i,j)\in P_1} d_{i,j} - \sum_{(i,j)\in P_2} d_{i,j} \right) \cdot r_1 + \left( \sum_{v_j\in P_1} v_j - \sum_{v_j\in P_2} v_j \right) \cdot r_2$$  \hspace{1cm} (2)
3. Provisioning Differential Delay Aware Survivable Service with Virtual Concatenation

Framer/mapper device contains internal or external memory for DD compensation. It can also specify a threshold for adding new member to an existing VCAT group. In a typical framer device [6], DD compensation is 250 µs using internal memory, and expandable up to 128 ms with off-chip SDRAM. How to provide optimized survivable service with respect to the differential delay constraints is a challenging problem. The DD-aware routing problem is NP-complete [8-9], and hence, heuristic algorithms are required. We find feasible paths first and then route the traffic and allocate the backup capacity. We state the problem as follows:

**Given:** An optical network, modeled by a graph G(V, E, C, g, DDC), where V is the node set, E is link set, and C is the capacity available on the link. Each link consists of multiple wavelength channels. Each network node supports VCAT granularity switching capability g, which can equal VT1.5, STS-1, STS-3c, etc. (High-Order or Low-Order VCAT). DDC is the differential delay constraint. We assume g is STS-1 in this study, and traffic can be switched between different wavelength channels and different fiber links. We have request R(s, d, B, holding_time), where B is the requested bandwidth, which is rounded to a multiple of the VCAT granularity g.

**Find:** Optimal survivable service between s and d with capacity B subject to the DDC. Service can be recovered in the case of any single link failure. Jointly optimize the network resources of working and backup bandwidth while maximizing the backup sharing.

3.1 Proposed Approaches: SPLIT

We propose two heuristic algorithms. Both are based on the idea of **Shared Protection of Largest Individual Traverse path (SPLIT)** but use different methods to calculate the paths between node pair. The first heuristic algorithm is based on K (K>=2) disjoint-paths algorithm [7], while the second one is based on min-cost flow algorithm. Actually, K is bounded by the minimum cut between the node pair. The disjoint-paths-based approaches can effectively avoid the trap topology [8-9]. If there are M paths between node pair for bandwidth B, we use (M-1) as working path, and assign path M as the backup with capacity as the maximum capacity of any working path for protection. Any working path can be re-routed to the backup path in case of failure. We also minimize the total reserved capacity by sharing the backup capacity of intra and inter VCGs. If there is no sharing, the resource overbuild for protection of B using M paths (assuming M > 2 and same cost for all paths) is 1/(M-1). If M = 1, we simply block the request. Intuitively, the bigger the M, the better it is for decreasing the resource overbuild. By using disjoint paths, we can maximize backup sharing since only disjoint working paths can share the backup capacity. Also we can get the benefit of simple management and operation. Given multiple paths, free capacity and cost between node pair, we still need to find an optimal scheme to provision the survivable service. Figure 1 shows an example of SPLIT, where \(p_w^1, p_w^2, p_w^3\) are three working paths, \(p_b\) is backup path, \(DDC_d\) is DD constraints of node d, \(p_b = \max(p_w^1, p_w^2, p_w^3)\), and maximum DD of \(\{p_w^{1-3}, p_b\} \leq DDC_d\).

**Algorithm 1:** SPLIT_KDP: K (K>=2) disjoint path based SPLIT.

1) Calculate all the K disjoint paths between node pair [7], using the cost function (Fig. 2) for load balancing, and pick the maximum number of feasible paths based on DDC constraints and Eqn. (2). \(d_e\) is the distance of edge e, \(W_e\) is number of total wavelength, and \(f_w\) is the free wavelength on edge e.

2) SPLIT (introduced above) the bandwidth B among all the “good” paths between node pair. Block the request if K<2.

3) Select the backup path between node pair and reserve capacity. Find the same source and destination pair inter-VCG sharing and reserve the difference of share capacity along the sharable path.

**Algorithm 2:** SPLIT_MCF: Minimum-Cost-Flow based SPLIT.

1) Set the link capacity as \(Bs = \left\lfloor \frac{B}{(MinCut(s, d) - 1)} \right\rfloor\).

2) Calculate the min-cost flow of total (B + Bs) based on the cost function of Fig. 2.

3) Select the maximum number of paths according to the DDC and Eqn. (2) and SPLIT. If any path is removed, increase Bs by 1 and repeat step 2) until Bs=B or (B+Bs) can be provided. Block the request if not assigned.

4) Optimize resource with sharing backup capacity. (Please note that in Step 3 the backup and working path don’t have to be disjoint.)
4. Illustrative Numerical Examples and Discussion
We simulated a dynamic network environment on the topology as shown in Fig. 3, with 16 wavelengths per link. We assume that the connection arrival processes are independent at all nodes and the connection holding time follows a negative exponential distribution. Connection requests are uniformly distributed among all node pairs. The number of connection requests follows the bandwidth (bps) distribution 50M : 100M : 150M : 200M : 250M : 500M : 10G = 100 : 50 : 20 : 10 : 10 : 4 : 2 : 1 (which is close to the bandwidth distribution of a practical network). The network load, in Erlang, is defined as connection arrival rate times average holding time times a connection's average bandwidth normalized in the unit of OC-192. For performance metrics, we use Bandwidth-Blocking Ratio (BBR), and Resource Overbuild (RO), which are defined in [8]. We set $r_1 = 5 \mu s/km$, $r_2 = 25 \mu s/node$ and assume all nodes have same DD compensation capability. Figure 4 plots BBR vs. load with or without DDC for SPLIT_KDP, and SPLIT_MCF. Figure 5 plots RO vs. load with or without DDC of SPLIT_KDP and SPLIT_MCF.

In the case of no DDC, SPLIT_KDP RO is 20%-25%, and is comparable to the minimum-cost-flow based SPLIT_MCF. Both are much better than existing heuristics in [8]. With the small constraints of DDC (8 ms or 10 ms), the efficiency benefits of VCAT will be negatively impacted. It cannot be ignored in the routing. However, when DDC is relatively large (50 ms), the performance is comparable or even better than the case without DDC in this example, since the very long path in the VCG group is removed in the routing with appropriate DDC.

Reference: